

- 1 -

**SYSTEM AND METHOD FOR BACK OFFICE PROCESSING OF  
BANKING TRANSACTIONS USING ELECTRONIC FILES**

JPB  
at

**FIELD OF THE INVENTION**

The present invention generally relates to systems and methods for  
5 back office processing of bank transactions and more particularly to a system and  
method which electronically captures transaction information at the time of  
execution of the transaction and uses this electronic information during the back  
office processing.

**BACKGROUND OF THE INVENTION**

10 Historically, a bank teller's interaction with customers and the  
bank's internal systems have been manually organized on a transaction by  
transaction basis with little to no capability of linking the information regarding  
the transaction with the paper(s) which constitute the transaction.

15 Some prior art teller systems maintained an electronic journal  
which allows the teller to perform limited research and allows the teller to reverse  
transactions selected from the electronic journal in the case where an error was  
actually discovered. However, this type of electronic journal is limited in its  
applicability as it is simply a sequential list of transactions performed at the bank.  
Furthermore, the electronic journal of the prior art has limited detail regarding  
20 transactions and must be supplemented by a continuous paper tape printed out at  
each teller system. Given the limited amount of information gathered at the teller  
workstation, the prior art methods and systems for back office processing were  
very labor and machine intensive. The term "back office" is well known to those

skilled in the art and relates to the facility in the back which performs the processing for the bank, e.g. posting of transactions, clearing of checks, statement generation ... Each branch of the bank forwards all of the documents (e.g., checks, deposits slips ...) to the back office at the end of the day for processing.

5 Back office processing for banking transactions includes for example, posting of transactions, statement processing, proof of deposit processing, end of day confirmation, account reconciliation and archiving. Until recently, most of the capture of data from the paper representing the transactions (e.g., checks) was done manually at the back office. For example, an operator would physically  
10 look at the check and enter the dollar amount written on the check into a record in the bank's database used for tracking checks. As larger banks process huge volumes of checks, each operator in the back office was responsible for data entry for thousands of checks per day. The sheer volume and repetitiveness of this process naturally led to errors in the data processing. Recently, systems have  
15 been developed which optically scan the financial documents data and use character recognition to capture the data previously captured by human operators (e.g., the amount written on a check). Fig. 1 illustrates such a prior art system and method of back office processing. Typically at the end of the day, each branch of the bank forwards all of the paper 10 associated with the day's  
20 transactions (e.g., checks, deposit slips ...) to a central location.

The first process undertaken at the back office is to capture both images of the paper (front and back) and the Magnetic Ink Character Recognition (MICR) data contained on the paper. Module 15, Check Processing Control System (CPCS) Prime Capture, accomplishes both of these functions using  
25 conventional image enabled sorters, optical readers and MICR readers. The image data of the two sides of each of the papers 10 is stored in an image archive database 20, while the MICR data read from the paper is stored in a CPCS database 25. Once the image and MICR data have been captured by the CPCS Prime Capture 15, a Character Recognition Engine 30 analyzes the captured

images in order to determine the amount of the transaction. If the Character Recognition Engine 30 successfully reads the amount of the transaction, the amount is used to update the record for that transaction in the CPCS database 25. Typically the Character Recognition Engine 30 is able to interpret the amount on  
5 approximately 60% of the transactions with a 2% error rate.

If the MICRline data on the paper document is read incorrectly, in the MICRline Data Completion module 35, an operator looks at the image of the document and manually completes the MICRline data in the transaction record for the document. If the Character Recognition Engine 30 fails to capture the  
10 amount of the transaction, in the Amount Key Entry module an operator manually read the image of the check from the image database 20 and inputs the amount into the transaction record contained in the CPCS database. If the character recognition for the amount on a deposit slip does not reconcile with the sum of the amount character recognized from the checks included on the deposit slip, the  
15 transaction balanced in the Deposit Balancing 45 module.

In Deposit Balancing 45, the amount listed on a deposit slip is compared with the total of the amounts of the checks associated with the deposit slip. If these two totals match, the deposit is considered balanced. If the totals do not match, an operator has to manually review the images of the deposit slip and the associated checks in order to determine and correct the error. Errors could  
20 occur in any number of areas such as incorrect character recognition by the Character Recognition engine or incorrect human input in the Amount Key Entry process 40.

As described above, the conventional back office processing of  
25 financial documents from the branches of the bank are very labor intensive and error prone. Even with the above described automation aids, the process is still very labor and machine intensive and still produces errors which can only be resolved by human intervention.

## SUMMARY OF THE INVENTION

In the present invention, as transactions are processed by a bank teller, the teller captures all of the relevant information with respect to the transaction for inclusion in an electronic file, the Electronic Journal. Each record  
5 in the Electronic Journal representing a transaction will contain at least the branch and teller identification, the type of transaction, any MICR data associated with the paper(s) constituting the transaction, and the amount of the transaction. In a preferred embodiment of the present invention, the Electronic Journals from all of the branches are forwarded periodically throughout the day to a central location  
10 associated with the branches. In addition to the creation of the Electronic Journal, each branch, at the end of the day, forwards all the paper associated with the transactions to the back office processing center. In conjunction with this end of the day processing, each branch creates an electronic summary of transactions processed that day at the branch (end of day sums).

15 In a preferred embodiment, a single processing center performs all of the back office processing for all branches of the bank. Periodically throughout the day, this single back office accesses the Electronic Journals for all of the branches. Alternatively, the back office processing center could poll each of the branches independently to access the Electronic Journals.

20 After the importation of the Electronic Journal into the back office, all of the transaction records from the Electronic Journal are stored in a Transaction Repository memory. In parallel with this storage of the transaction records, a subset of the information contained in each of the transaction records is forwarded the back office module which performs the End of Day Confirmation  
25 process.

When the paper transactions arrive at the back office from the various branches, each of the paper transactions is imaged in the conventional manner and the MICR data is read from the paper. The back office can now automatically correlate the images and MICR data captured from the paper with

the complete transaction record contained in the Transaction Repository. Since the transaction record already contains all of the information relevant to the transaction, the need to perform the character recognition, MICRline data completion and amount key entry performed in the prior art is eliminated (except in exception cases).

The present invention has numerous advantages over the prior art processing methods and systems including: a reduction of labor and equipment; a reduction in errors associated with the character recognition process; a reduction in the amount of character recognition and manual key entry which has to take place; simplification of the teller process due to far fewer sorts of physical paper; enabling of a more constant flow of work from the branches to the back office; extended customer hours for centralized Automated Teller Machines (ATM) and branch processing due to freed capacity at the back office; end of day proofing by tellers is all but eliminated; easy identification of missing papers; enhance the return items process for the bank; enable accurate assignment of the float for cashed checks; reduction of "on-us" financial control documents (e.g., General ledger tickets, Cash-in, Cash- out); and an improvement of the cash reconciliation process based upon earlier access to the document images.

## BRIEF DESCRIPTION OF THE DRAWINGS

For the purposes of illustrating the present invention, there is shown in the drawings a form which is presently preferred, it being understood however, that the invention is not limited to the precise form shown by the drawing in which:

Figure 1 illustrates the back office processing of the prior art; Figure 2 depicts a typical hardware configuration at a branch location;

Figure 3 illustrates the three groups of transactions which are generated at a branch;

Figure 4 is an overview of the system and the flow of data according to the present invention;

5                   Figure 5 illustrates the process conducted by the Import module;

Figure 6 illustrates the detail of the End of Day Confirmation module and associated processes;

Figure 7 illustrates the detail the Work In Progress component;

10                   Figure 8 depicts an overview of the process of linking the electronic transactions in the Transaction Repository 175 with the data captured by the CPCS Prime Capture process;

Figure 9 illustrates the process conducted by the Update component;

15                   Figure 10 depicts the process by which the Archive Cross Reference Builder creates an Archive Cross Reference database;

Figure 11 depicts the process performed by the All Items File Builder; and

Figure 12 illustrates the deposit balancing procedure.

## DETAILED DESCRIPTION OF THE INVENTION

20                   As described above, part of the present invention involves the teller capturing information concerning a transaction as the transaction is being processed by the teller. Fig. 2 illustrates a typical hardware configuration at a branch which includes a token ring network 50 of workstations. One of these networks 50 is established at each branch of the bank. Elements 55 represent  
25                   workstations for use by the tellers in conducting transactions (e.g., at a teller window with a bank customer). Each teller workstation 55 also includes a MICR

reader 60 which enables the teller to scan the MICR encoding on checks and other instruments.

Server 65 is an application server, while server 70 is a database server. Each of these servers 65, 70 provide common access to all of the applications and databases used at the branch. Through the network 50, each of the teller workstations 55 have access to the applications and databases residing on servers 65, 70. The application server 65 is shown as having a shared laser printer 75 connected thereto. Server 80 is a communication server which provides the connection between the network 50 and the branch host 85. As will be described below, transactions generated from the teller's workstations 55 are transmitted to the branch host 85 for confirmation processing.

A logical grouping of one or more consecutive transactions performed by a teller is known as a session. A customer session is one or more transactions performed for a single customer during the course of a single visit to the teller window. A non-customer session consists of one or more transactions performed by the teller without a customer being physically present at the teller's window.

As a teller initiates a session, a new record is opened in the Branch Electronic Journal which includes the number and date and start time of the session. The Branch Electronic Journal is, as implied by its name, a file which electronically records all of the actions (transaction processing) performed by the teller at the branch. Each session record in the Branch Electronic Journal includes the session ID and session start and end times and initial Session Balance (which may be zero), along with the Teller ID of the teller conducting the session and any customer ID obtained. Each transaction processed by the teller has a separate record entry within the session record in the Branch Electronic Journal.

The transaction entry contains all pertinent details regarding the transaction including, for example: a transaction sequence number; the transaction type; account number; and the dollar amount. There are

approximately two hundred and fifty different types of transactions which can be processed, but the back office is only concerned with approximately one hundred and thirty. In one embodiment of the present invention, the branches 150 only send the back office 170 transactions of the pertinent one hundred and thirty types. In an alternative embodiment, the branches 150 send records to the back office 170 related to all of the transactions conducted at the respective branches 150, regardless of type. In this embodiment, the back office 170 culls out only the types of transactions which require back office processing.

Each transaction is logged in the Branch Electronic Journal as it is sent to the branch host system 85 for confirmation. The transaction entries in the Branch Electronic Journal therefore reflect the "stream" order in which the transactions are sent to the branch host 85. The branch host response for each transaction is appended to the appropriate transaction entry as received, along with the time of receipt. The Branch Electronic Journal may be viewed and sorted selectively by the teller, but may not be edited or altered by any user, but may be selected by the teller for reversal. All teller activity is permanently recorded in the Branch Electronic Journal.

For each transaction which involves a piece of paper with a MICR code imprinted thereon (e.g., a check), the teller system prompts the teller to swipe the check(s) on reader 60 to capture the check information. The teller then individually enters the amount related to the paper (e.g., the amount of each check). This MICR data is appended to the transaction record in the Branch Electronic Journal.

As briefly described above, there are over two hundred and fifty different types of transactions which can be conducted at a branch. The back office processing is only concerned with approximately one hundred and thirty of these transactions. These one hundred and thirty transactions are logically and physically separated into three different groups. Fig. 3 illustrates the three



different groupings, Group 1 (100), Group 2 (105) and Group 3 (110) of transactions generated at the various branches of the bank. The physical separation of the groups occurs at the teller's workstation when the transactions are processed. At the beginning of the day, or any point throughout the day, the teller generates three control tickets 115 which identify the group of transaction papers 120 which are physically grouped with the control ticket 115. As the teller processes transactions, the physical paper 120 associated with the transactions is separated into the one of the three groups defined by the control tickets 115. Periodically throughout the business day, and at least at the end of the branch day, all of the paper 115, 120 associated with the transactions processed by the branch, organized by the three groups, is shipped to the back office for processing as described below.

Group 1 (100) consists of credits and debits with cash, check or credit offsets and includes cash only deposits. The Group 1 (100) transactions are financially complete at the teller's window, and no further financial processing is required by the bank. The back office 170 requires the documentation with respect to these transactions for at least archival and research purposes. For cash only deposits, the MICR encoded deposit slip is the paper associated with the transaction. Group 2 (105) transactions are all deposits except cash only deposits and include check only, mixed, split and check deposits less cash. In processing Group 2 transactions, the teller swipes the deposit slip 120 in order to acquire the MICR data (e.g., the account and serial numbers). The teller further enters the cash amount (if any) and the total amount of the deposit. For each check in a Group 2 (105) deposit, the teller MICR swipes the check 120 and manually enters the dollar amount of the check. In a preferred embodiment of the present invention, the tellers only swipes and enters the amounts of Group 2 (105) checks 120 if the total number of checks involved in the transaction is below a threshold amount (e.g., less than five checks). If there are more checks 120 involved in the

transaction than the threshold amount, the traditional back office processing described above is used to process the transaction at the back office.

Group 3 (110) transactions consist solely of checks 120, either cashed or used for payments (e.g., of a credit card balance) or sales (e.g., of travelers checks). As with checks 120 in Groups 1 and 2 (100, 105) checks 120 for Group 3 (110) are MICR swiped by the teller and the teller manually enters the dollar amount. As described above, all of the MICR data and the dollar amounts manually entered by the teller are input into record in the Branch Electronic Journal associated with the transaction. This is true for all transactions, whether they be Group 1, 2 or 3. Group 2 (105) and Group 3 (100) transactions are considered "live" even after they leave the branch 150 since the financial processing for the transaction is not yet complete (e.g., clearing of checks through the bank upon which the check is drawn).

Fig. 4 illustrates an overview of the system and the flow of data according to the present invention. As previously described, the various branches 150 of the bank forward their individual Branch Electronic Journals to a retail mainframe system 155 common to all of the branches 150. The forwarding of data is accomplished, for example, through a telecommunications line. Note that the term retail relates to the branch operations of the bank. Again, the Branch Electronic Journals contain records related to over two hundred and fifty different types of transactions and also contain other branch specific information which is of no interest to the back office. Accordingly, the retail mainframe 155 extracts only the information which is of interest to the back office (relating to approximately one hundred and thirty transactions) and generates a Transaction Journal file 160 which is for use by the back office.

As transaction records are received by the Retail mainframe 155 throughout the course of the day, the mainframe 155 builds the Transaction Journal 160. The process of building the Transaction Journal 160 is invoked at

the start of each business day, and as described below passes certain transactions contained in the Transaction Journal 160 to the back office platform on a continuous basis throughout the day. As electronic transactions are received from branches 150 in the Branch Electronic Journals, the transactions are examined to determine if they are to be written to the Transaction Journal 160. By the end of the day, the Transaction Journal file 160 contains records for all of the relevant transactions performed at all of the branches 150 for the relevant time period (e.g., a day).

The following Transaction types are included in the Transaction Journal 160: Financial transactions (e.g., Group 2 (105) Deposit Tickets and associated checks, and Group 3 (110) Payment, Sale, or Cashed Checks that have been MICR Swiped by the Teller); Archive Only transactions (e.g., Group 1 (100) transactions); Work In Progress Batch (Groups 1, 2, and 3) transactions; Branch Confirmation transactions; and All Items Only transactions (e.g., non paper transactions). The relevancy of these different types of record will be discussed below with respect to the back office processing.

The records contained in the Transaction Journal 160 preferably include the following fields: Financial Entity; Branch Number; Business Date; End Of Day Indicator; Transaction Type Indicators; Transaction Count; Transaction Dollar Amount; Discrepancy Indicator; Confirmation Number; Confirmation Time; Session Number; Teller ID Number; and the MICR line data captured at the teller station.

As depicted in Fig. 4, an Import module 165 of the present invention is located at the back office 170. The Import component 165 processes transactions from the Transaction Journal 160 which it reads throughout the course of the day. The Import component 165 is invoked at the start of each business day and executes (performs the read operation) periodically throughout the day (e.g., every 15 minutes) in order to pick up new transactions. The import

module 165 examines each transaction that has been imported and invokes the appropriate back office 170 components that are required to process the transaction.

Each transaction imported during a given import run is examined to determine the type of the transaction. If the transaction is a Financial transaction, as described above, the record for that transaction is stored in the Transaction Repository 175. If the transaction is for archive purposes only (e.g., Group 1 transactions) the transaction is passed to an Archive Cross Reference Builder component 180 (see Fig. 10). If the transaction reflects a Work In Progress transaction, the transaction is passed to a Work In Progress component 185. Branch Confirmation transactions are passed to an End Of Day Confirmation component 190. Finally, if the transaction is an All Items Only transaction (e.g., a non paper transaction) it is stored in the Transaction Repository 175.

The Import component 165 is invoked at the start of each business day either by the System Manager 195, or by another means of job scheduling. Subsequent import runs are invoked automatically (e.g., every 15 minutes). The System Manager 195 provides an On-line means of invoking the daily start-up of the Import process. The System Manager 195 further allows the operator to monitor and control the Import process by viewing the result of each import run. The System manager 195 also allows monitoring and control the End of Day Confirmation process between the back office 170 and the Retail mainframe 155, and provides for the management of system exception conditions which may occur within the back office 170. Back office processing exception conditions are stored in a system journal (not shown) and can be viewed via a System Manager 195 screen. Exception conditions are detected by each component of the back office platform 170 and are recorded in the system journal. The System manager 195 also allows for viewing of Work In Progress Batches that have been

“cut” by a given teller (as described below). The files required to view this data are created by the Work In Progress component 185.

Fig. 5 illustrates the process conducted by the Import module 165. As previously described, the import module 165 is invoked at the beginning of the day and is periodically invoked (e.g., every fifteen minutes) in order to pick up new transactions. For each record in the Transaction Journal 160, Import 165 tests to see what should be done with the record in steps 200, 210, 220, 230 and 240. If the answer to any of the tests is NO, the testing of the transaction goes onto the next test. Import 165 uses a control file 162 in order to keep track of the status and progress of the transaction processing and to store control parameters. As previously described, the System Manager 195 is used to invoke and monitor the import process.

In step 200 it is determined if the transaction is a Group 1, 2 or 3 transaction. If the transaction is a Group 1, 2 or 3 type transaction, the transaction is stored in the Transaction Repository 175. In step 210 it is determined if the transaction is a branch confirmation transaction (e.g., end of day totals). If the transaction is a branch confirmation, the confirmation is passed onto the End of Day Confirmation module 190. If the transaction is a Work in Progress transaction (step 220) the transaction is passed onto the Work in Progress module 185 (step 225). If the transaction is an adjustment transaction (step 230) the transaction is passed onto an Adjustment module (step 235). Adjustment transactions are entered by a teller to correct a previously submitted transaction. The adjustment to the original transaction can be done either manually by an operator or automatically by the system. Finally, if the transaction is an All Items Only transaction (step 240) the transaction is stored in the Transaction Repository 175 (step 245).

The Transaction Repository 160 is one of the key elements of the system and method of the present invention. As previously described, electronic

data concerning transactions which was previously captured at the back office 170 using expensive, time consuming and error prone processes and equipment is now captured by the tellers at the time of the transaction, imported and stored in the Transaction Repository 160. The Transaction Repository 160 provides the primary data storage, indexing, and retrieval services for the present invention. This Repository 160 provides for concurrent data storage, indexing, and retrieval by the other modules and processes described herein.

Each transaction resident in the Transaction Repository 160 is indexed at least by the MICR data provided by the teller MICR swiping process. The transactions are preferably further indexed by the Branch, Teller and Batch (group 1, 2 or 3) in order that the other modules of the system can quickly locate and access the transaction data. As previously described, transactions are passed from Import 165 via an application program interface (API) to the Transaction Repository 175. The transactions stored in the Transaction Repository 175 include: Financial transactions (Group 2 Deposit Tickets and associated checks and Group 3 payment, sales, or cashed checks, that have been MICR Swiped by the Teller); Archive Only transactions (Group 1 transactions); and All Items Only transaction (non paper transactions).

Fig. 6 illustrates in more detail the End of Day Confirmation module 190 and associated processes. The primary purpose of the End of Day Confirmation module 190 is to confirm, at end of the branch day, the completeness of the data feed from the branches 150 to the back office 170 through the Transaction Journal file 160 (see Fig. 4). As depicted in Fig. 6, the End of Day Confirmation module 190 maintains an End of Day Confirmation File 192. This file 192 contains at least one record for each branch 150 of the bank. Each of these branch records contain accumulators which keep track of the financial totals processed by the branch during the course of the entire day. The accumulators are organized by the transaction types (e.g., cash only deposits).

As transactions are received by Import 165, the transactions are passed to the End of Day Confirmation module 190. The End of Day Confirmation module 190 reads the branch from the transaction record and uses this information to retrieve the branch record from the End of Day Confirmation File 192. The dollar amount related to the transaction is then aggregated within the branch record with respect to the transaction code type.

At the end of day in each branch, Confirmation transactions are passed to the End of Day Confirmation component 190 from the Import component 165. There are confirmation transactions for each transaction code type for each branch. The confirmation process in the End of Day Confirmation component 190 matches transaction code values to ensure that all work has been successfully received by the back office 170. This is accomplished by comparing the values in the received Confirmation transactions with the values accumulated throughout the day in the branch records contained in the End of Day Confirmation File 192.

If the confirmation process fails for a given branch, this failure is displayed on the System Manager 195 screen. This notification provides an early warning to the back office personnel that a potential problem exists in the teller, branch or back office systems.

Fig. 7 illustrates in more detail the Work In Progress component 185 of the present invention. A work in progress consists of one or more "live" transactions which have not been completely processed. The primary function of this component 185 is to write Work In Progress transactions to a Work In Progress data set 187 as the transactions are received by the back office 170. The purpose of this function is to enable the tracking of these transactions as they proceed through the various processing steps and stations at the back office 170. The Work In Progress data set 187 is a repository where units (batches) of Work in Progress are recorded. Although not shown in Fig. 7, the Work In Progress

data set 187 is accessible by the System Manager 195 for monitoring and tracking of Work In Progress transactions.

As illustrated in Fig. 7, Work In Progress transactions are fed to the Work In Progress component 185 by both the Import component 165 and the End of Day Confirmation module 190. Additionally, sensors can be placed throughout the back office 170 and the branches 150 (see Fig. 4) in order for these components to report the location and progress of a Work In Progress. These sensors are generically represented in Fig. 7 as Other Modules 197. For example, as the physical paper arrives at the receiving office of the back office 170 from a branch 150, an operator scans the control card (115, Fig. 3) on top of the batch. This sensor feeds the control information for the batch to the Work In Progress module 185 for inclusion in the Work In Progress database 187. Similarly, as the batch is fed into the sorter (not shown), a sensor in the sorter reads the control card 155 and informs the Work In Progress module 185 which updates the record for the batch in the Work In Progress database 187. By this means, the system is able to track the location and progress of the transactions as they pass through the back office 170.

The power of the present invention is readily apparent once all of the paper 10 representing all of the transactions processed throughout the day at the branches 150 arrives at the back office 170 from the branches 150. Fig. 8 depicts an overview of the process of linking the electronic representations of the transactions contained in the Transaction Repository 175 with the data in the Image Archive 20 and the CPCS database 25 generated by the CPCS Prime Capture process 15.

As previously described with respect to the prior art system of Fig. 1, the first process undertaken at the back office 170 of the present invention is to capture both images of the paper 10 (front and back) and the MICR data contained on the paper 10. As with the prior art, CPCS Prime Capture 15



accomplishes both of these functions using conventional image enabled sorters, optical readers and MICR readers. The image data of the two sides of each of the papers 10 is stored in an image archive database 20, while the MICR data read from the paper is stored in a CPCS database 25. This is where the similarity to the prior art systems and methods ends. As previously described, the prior art system of Fig. 1 employed: a Character Recognition Engine 30 to analyze the captured images in order to determine the amount of the transaction; a MICRline Data Completion module to [NOTE to inventors, what does (did) this module do?]; and if either of the automated processes failed to capture the amount of the transaction, an operator would have had to manually read the image of the check from the image database 20 and input the amount into the transaction record contained in the CPCS database (module 40).

In direct contrast to the systems and method of the prior art, the present invention uses an automated Update component 200 which is used to link the data electronically captured by the teller and contained in the Transaction Repository 175 with the data contained in both the Image Archive 20 and the CPCS database 25. In general, the Update component 200 extracts the relevant information (e.g., dollar amount or account number) related to a transaction from the Transaction Repository 175 and inserts this information into the record for paper 10 associated with the transaction contained in the CPCS database 25. The CPCS database 25 already has a pointer to the image of the paper 10 in the Image Archive 20. In this manner, all of the relevant information regarding transactions is gathered and is easily accessible without the expensive and time consuming processes and equipment of the prior art.

As described above, the primary function of the Update component 200 is to obtain transaction specific data from the Transaction Repository 175 following the completion of a CPCS Prime Pass Capture 15 run. Update module 200 updates the CPCS database 25 with dollar amounts for checks, account

numbers for Counter Deposit Tickets, and Cash In/Out Amount for mixed deposits. As previously described, the data for the transaction contained in the Transaction Repository 175 is generated at the teller workstation at the branch 150. In contrast, in the prior art, this type of data was either manually entered by a clerk at the back office 170 or was attempted to be captured by the imaging system. As described above, both of the prior art systems were error prone and very labor intensive. As the teller is able to swipe the documents and enter the financial data at a more deliberate pace, the likelihood that errors would occur during the use of the present invention are. Furthermore, as the teller is directly responsible for the integrity of each financial transaction he or she processes the teller presumptively double checks each transaction. As this verification by the teller occurs "offline" with respect to the back office processing, the speed of the back office processing is greatly enhanced. Finally, the teller system includes an automatic proofing function which electronically catches errors (e.g. in deposit slip or check amount entry) by the teller before they enter the system.

Update component 200 is invoked after the completion of a CPCS Prime Pass Capture 15 on branch work (i.e., physical paper 10 of Group Types 1, 2 and 3). The paper 10 associated with Group 2 & 3 type transactions is captured throughout the course of the evening and paper related to Group 1 type transactions are captured at end of day. The output of the CPCS Prime Pass Capture 15 is a string which contains records for all of the paper associated with the transactions processed during the run. The string is stored in the CPCS database 25.

Fig. 9 illustrates the process conducted by the Update component 200 after completion of the CPCS Prime Pass Capture 15. The first operation is that a Read and Update module 210 of the Update component 200 opens the string stored in the CPCS database 25 in order to read the transactions. The transactions are then read one by one. When a Group 1, 2 or 3 type transaction is

encountered on the string, the Read and Update module 210 reads from the string the MICR data for the transaction. After the Read and Update module 210 has read the MICR data for the transaction from the string in the CPCS database 25, a Get module 205 of the Update component 200 accesses the Transaction Repository 175 using the MICR data as an index to search the Transaction Repository 175. As depicted in Fig. 9, the Transaction Repository 175 is comprised of an Index 176 and a Detail database 177. As previously described, the transactions stored in the Transaction Repository 175 are indexed in several ways in order to ease the access of the detailed transaction data. These various indexes are stored in the Index database 176 while the complete record for the transaction is stored in the Detail database 177. The MICR data is used as an index key to search the Index database 176 which is then used to retrieve the transaction specific data (e.g., dollar amount for a check, account number for a counter deposit ticket, or Cash In/Out amount for a mixed deposit) from the Detail database 177.

The Transaction Repository 175 returns the retrieved transaction data to the Get module 205 which passes the data onto the Read and Update module 210. The Read and Update module 210 then updates the transaction data contained in the string in the CPCS dataset 25 with the retrieved data (e.g., the check dollar amount). For Counter Deposit Tickets, the dollar amount and account number are inserted into the CPCS string. For deposits with Cash In/Out, a Cash In/Out transaction will be inserted into the CPCS string.

After the Update component 200 has read all of the transactions from the string, retrieved all of the data from the Transaction Repository 175 and updated the string, the Update component 200 passes to the Work In Progress component 185 (Fig. 7) an electronic list of all work in progress batch numbers present on the string in order to track the progress of the transactions.

Group 1 transactions, credits and debits with cash, check or credit offsets and cash only deposits require special processing in order to track these transactions. As previously described, Group 1 transactions are no longer considered "live" as the financial aspect of the transactions has been completed at the teller work station. As such, Group 1 transactions are treated differently than Group 2 or 3 transactions which require financial processing. Fig. 10 illustrates the process by which the Archive Cross Reference Builder 180 creates an Archive Cross Reference database 215. The primary function of the Archive Cross Reference Builder 180 is to create the electronic Archive Cross Reference file 215 for transactions in Group 1, Archive Only in order to create the connection between the representation of the Group 1 financial transactions represented by the records in the Transactions Repository 175 and the records for the transactions contained in the CPCS dataset 25. As an alternative to the Archive Cross Reference Builder 180, the Update module 200 could perform this cross reference function. It is preferred though, to have the separate Archive Cross Reference Builder 180 perform the cross referencing as this function can be performed out the critical path of the Update module 200 which is also processing the "live" financial transactions of Groups 2 and 3.

The physical documents associated with on-line transactions (Group 1) created at the teller workstation 55 (Fig. 2) are delivered to the back office 170 where the MICR and image data is captured by the CPCS Prime Pass Capture component 15 as described above with respect to Fig. 8. The string containing the transactions resulting from the CPCS Prime Pass Capture 15 run is stored in the CPCS database 25. Similar to the process described above with respect to the Update component (Fig. 9) the Archive Cross Reference Builder 180 uses the MICR data as an index to search the Transaction Repository 175. The MICR data is used as an index key to search the Index database 176 which is then used to retrieve the transaction specific data (e.g., dollar amount) from the Detail database 177. With the electronic transaction information from the

Transaction Repository 175 and the captured transaction information from the CPCS database 25, the Archive Cross Reference Builder 180 generates a cross reference record which is written to the Archive Cross Reference file 215. In a preferred embodiment of the present invention the cross reference comprises pointers to the appropriate records in the CPCS database and the Transaction Repository. The Archive Cross Reference file 215 is used by the conventional back office processes such as research and adjustment programs.

Not all of the transactions conducted by the teller generate paper documentation which is sent to the back office 170. Accordingly, the present invention provides an All Items File Builder 250 as depicted in Fig. 11 in order to record these paperless transactions. As previously described, all of the transactions are electronically captured at the teller workstation 55 (Fig. 2) and are populated into the Transaction Repository 175. The ultimate function of the All Items File Builder 250 is to pass the electronic data regarding paperless transactions to the conventional back office processes (e.g., research and adjustment platforms).

The All Items File Builder 250 requests from the Transaction Repository 175 all of the transactions which are electronic, no paper transactions. The Transaction Repository is able to identify and extract these records and pass them back to the All Items File Builder 250. Each of these records is written to a No Paper database 255. Each of the records is read from the No Paper database 255 by another module (not shown) and the records are then used to create new records in the CPCS database 25 which had previously only contained records for transactions captured by the CPCS Prime Pass Capture 15 (Fig. 8). The CPCS database 25 now has a complete set of data relating to all of the transaction processed at the branches throughout the day.

Fig. 12 illustrates the Deposit balancing function as implemented in the present invention. The process of Deposit balancing in the present invention

is essentially the same as described above with respect to the prior art of Fig. 1. One big difference is that because of the data entry by the teller and inclusion of this data in the CPCS dataset (by Update module 200, Fig. 9) a drastically reduced number of deposit transactions will not be automatically balanced by the system. Accordingly, fewer deposit transactions will require manual balancing by an operator. A further improvement of the present invention is the case of missing documents, the teller data contained in the Transaction Repository 175 can be used to supplement the data scanned from the physical documents (contained in CPCS database 25, Fig. 8).

10 In step 500 in Fig. 12, a record or a group of records comprising a transaction is read from the CPCS database 25 (Fig. 8). Use of the data from the CPCS database 25 is preferred to balancing from the teller data found in the Transaction Repository 175 (Fig. 4) as this serves as a double check of the balancing performed by the teller at the teller's workstation. In step 510 it is tested whether or not the transaction is a Group 2 or 3 transaction. If the transaction is not a Group 2 or 3 type (i.e., it is a Group 1 transaction), the next transaction is read (step 500) from the Transaction Repository 175. Group 1 transactions do not require balancing as the financial aspect of the Group 1 transactions is complete at the teller's workstation.

20 If the transaction is either a Group 2 or 3 type transaction, the deposit slip (for Group 2) or batch ticket (for Group 3) record is read for the transaction (step 520). The amount of the transaction is read from the deposit slip or batch ticket record and written to a first temporary buffer (step 530). In step 540 the records for each of the documents (e.g., checks) which are associated with the deposit slip or batch ticket for the transaction are read. The dollar amounts of all of the documents in the transaction are summed in step 500 and the total is written to a second temporary buffer. In step 560, the first and second buffers are compared to verify that the transaction is in balance. If the two

buffers are equal, the transaction is balanced (step 570) and the process (steps 500-570) is repeated for the next transaction.

If the two buffers are not equal, the deposit is out of balance and requires human intervention. In step 580 the records for the transaction from the Transaction Repository 175 and the CPCS database 25, and the images of the documents representing the transaction (from the image database 20, Fig. 8) are displayed to an operator. Given all of this information on a single workstation/display screen, the operator is then able to determine why the deposit did not balance.

If a record for a document is missing from the CPCS database 25 and from the image database 20, but a record exists in the Transaction Repository, the operator performing the balancing has the option of accepting the dollar amount which appears from the Transaction Repository 175 and approving the balancing. In this case, the document could be missing, for example, if the document was misfiled with a different batch or was not read properly by the CPCS prime pass operation (e.g., was physically behind a different document). Alternatively, the system itself may balance an out of balance transaction under certain conditions (e.g., one check is missing, and the record for that check in the Transaction Repository 175 indicates the amount of the check is below a threshold amount (e.g., \$100)).

While the operator is investigating out of balance transactions, the process steps 500-570 run in parallel on the remaining of unprocessed transactions in the transaction repository. Out of balance transactions may be buffered for processing by the operator.

As previously described, the system and method of the present invention provides significant advantages over the prior art processing methods and systems. Less manual labor and less expensive equipment is required to implement and run the system of the present invention. The present invention

experiences a significant reduction in errors associated with both the character recognition process and the manual processes of the prior art. Although the teller is electronically capturing more data than in the past, the present invention simplifies the teller process due to far fewer sorts of the physical paper associated with the transactions. The system and method of the present invention enables a more constant flow of work from the branches to the back office which in part results in extended customer hours for centralized Automated Teller Machines (ATM) and branch processing due to freed capacity at the back office. The present invention all but eliminates the end of day proofing previously performed by tellers is all but eliminated.

The system of the present invention provides for easy identification of missing papers and enhances the return items process for the bank. Using the present invention, accurate assignment of the float for cashed checks is enabled. The system reduces the number of "on-us" financial control documents (e.g., General ledger tickets, Cash-in, Cash- out). The system further improves the cash reconciliation process based upon earlier access to the document images.

Although the present invention has been described in relation to particular embodiments thereof, many other variations and other uses will be apparent to those skilled in the art. It is preferred, therefore, that the present invention be limited not by the specific disclosure herein, but only by the gist and scope of the disclosure.